dortmund university
computational statistics

Normalizing Flows for Simulation-Based Expert Prior Elicitation

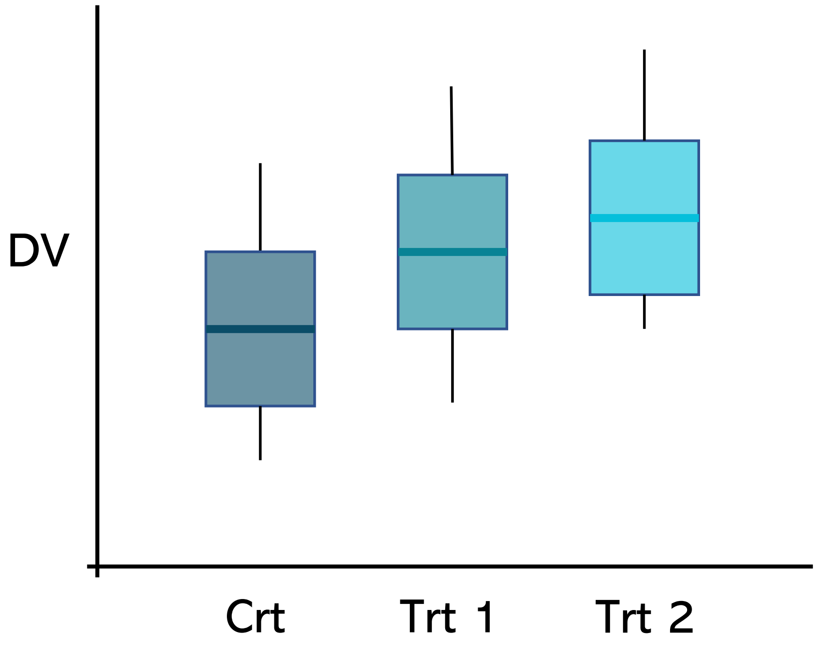
Florence Bockting
Stefan T. Radev
Paul-Christian Bürkner

MATHPSYCH 2024

Motivation

One factorial design with three levels

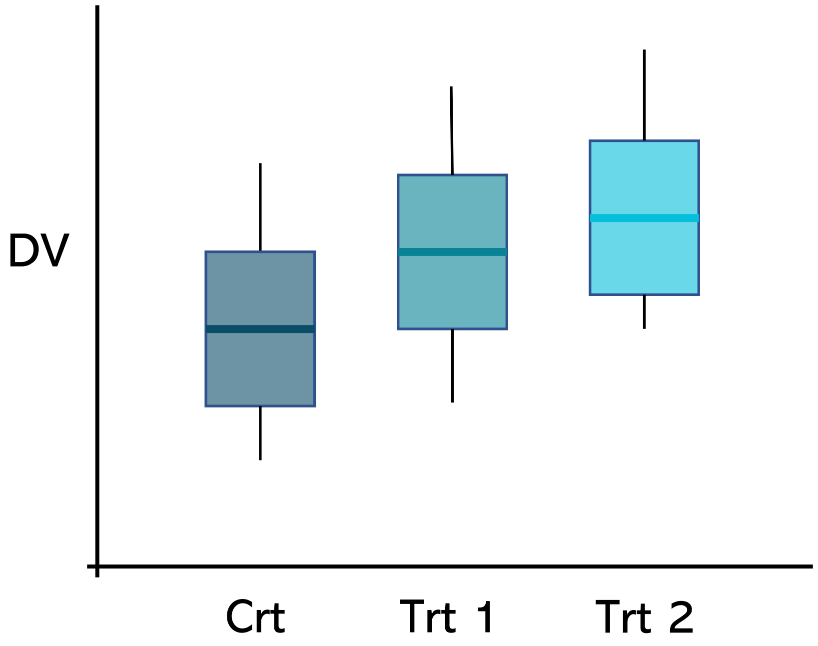
- One-factorial design with one three-level factor and a continuous dependent variable
- Levels:
  - control
  - treatment 1
  - treatment 2



Motivation

One factorial design with three levels

- One-factorial design with one three-level factor and a continuous dependent variable
- Levels:
  - control
  - treatment 1
  - treatment 2



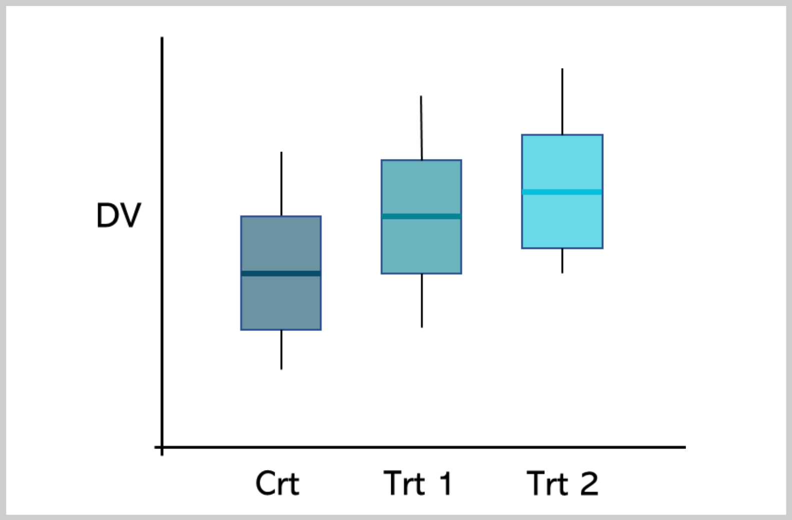
$$(\beta_0, \beta_1, \beta_2, \sigma) \sim \text{?}$$

$$\mu_i = \beta_0 + \beta_1 x_{1,i} + \beta_2 x_{2,i}$$

$$y_i \sim \text{Normal}(\mu_i, \sigma)$$

The problem

Translate expert beliefs into corresponding priors



$$(\beta_0, \beta_1, \beta_2, \sigma) \sim ?$$

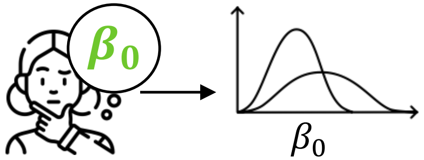
$$\mu_i = \beta_0 + \beta_1 x_{1,i} + \beta_2 x_{2,i}$$

$$y_i \sim \text{Normal}(\mu_i, \sigma)$$

The problem is actually not new

Translate expert beliefs into corresponding priors

- Traditionally, focus on parameter space

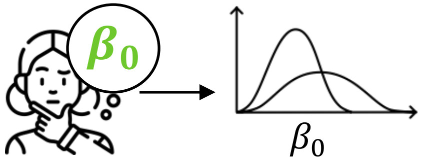


(see Mikkola et al., 2024 for comprehensive review)

The problem is actually not new

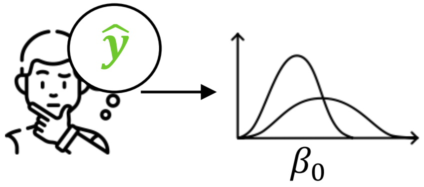
Translate expert beliefs into corresponding priors

- Traditionally, focus on parameter space



(see Mikkola et al., 2024 for comprehensive review)

- Recently, focus on prior predictive distribution (observable space)



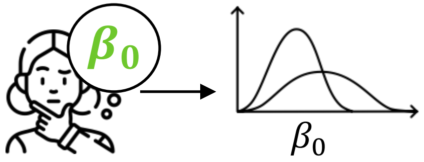
(e.g., da Silva et al., 2019; Hartmann et al., 2020; Manderson & Goudie, 2023; Perepolkin et al., 2024)

The problem is actually not new

Translate expert beliefs into corresponding priors

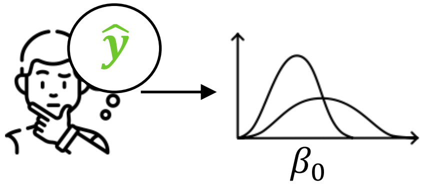
tu dortmund university
computational statistics

- Traditionally, focus on parameter space



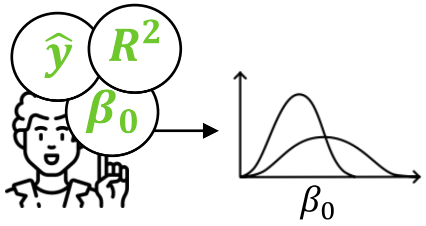
(see Mikkola et al., 2024 for comprehensive review)

- Recently, focus on prior predictive distribution (observable space)



(e.g., da Silva et al., 2019; Hartmann et al., 2020; Manderson & Goudie, 2023; Perepolkin et al., 2024)

- Focus on parameter space, observable space, and derived quantities



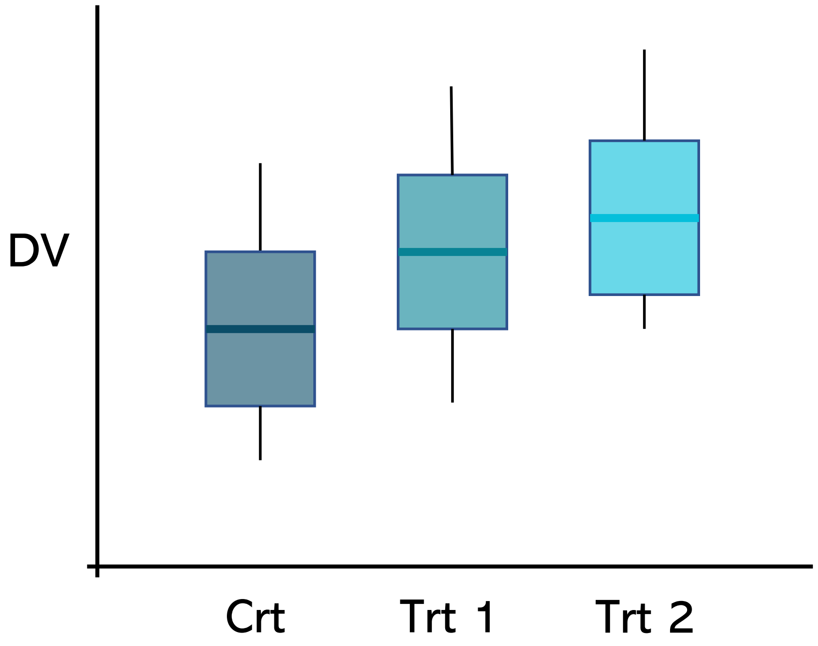
Bockting, Radev, & Bürkner (2023)

florence.bockting@tu-dortmund.de

Revisited: Motivation

One factorial design with three levels

- One-factorial design with one three-level factor and a continuous dependent variable
- Levels:
  - control
  - treatment 1
  - treatment 2



λ

$(\mu_0, \sigma_0, \mu_1, \sigma_1, \mu_2, \sigma_2, \alpha, \beta) = ?$

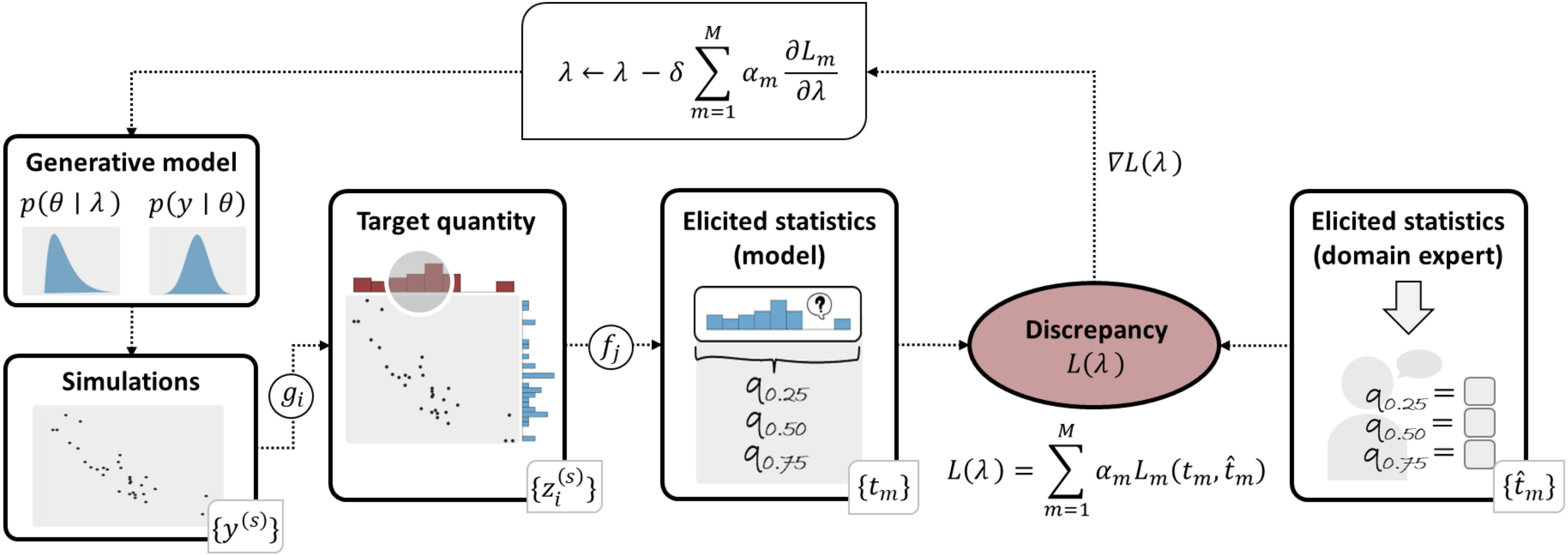
$$\beta_0 \sim \text{Normal}(\mu_0, \sigma_0)$$
$$\beta_1 \sim \text{Normal}(\mu_1, \sigma_1)$$
$$\beta_2 \sim \text{Normal}(\mu_2, \sigma_2)$$
$$\sigma \sim \text{Gamma}(\alpha, \beta)$$

$$\mu_i = \beta_0 + \beta_1 x_{1,i} + \beta_2 x_{2,i}$$

$$y_i \sim \text{Normal}(\mu_i, \sigma)$$

What we did and…

Bockting, Radev, & Bürkner (2023)

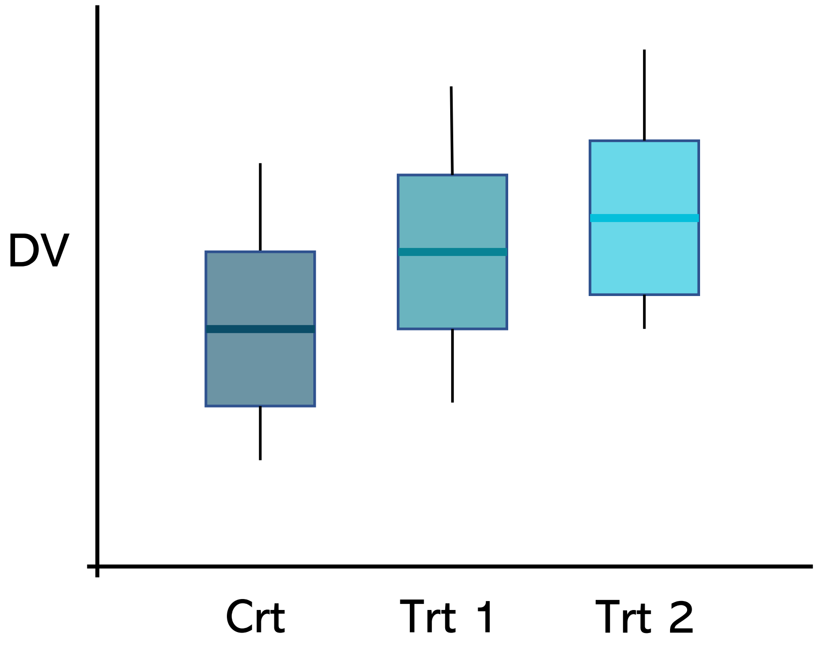


florence.bockting@tu-dortmund.de

Revisited (once more): Motivation

One factorial design with three levels

- One-factorial design with one three-level factor and a continuous dependent variable
- Levels:
  - control
  - treatment 1
  - treatment 2



$$(\beta_0, \beta_1, \beta_2, \sigma) \sim p_\lambda(\cdot)$$

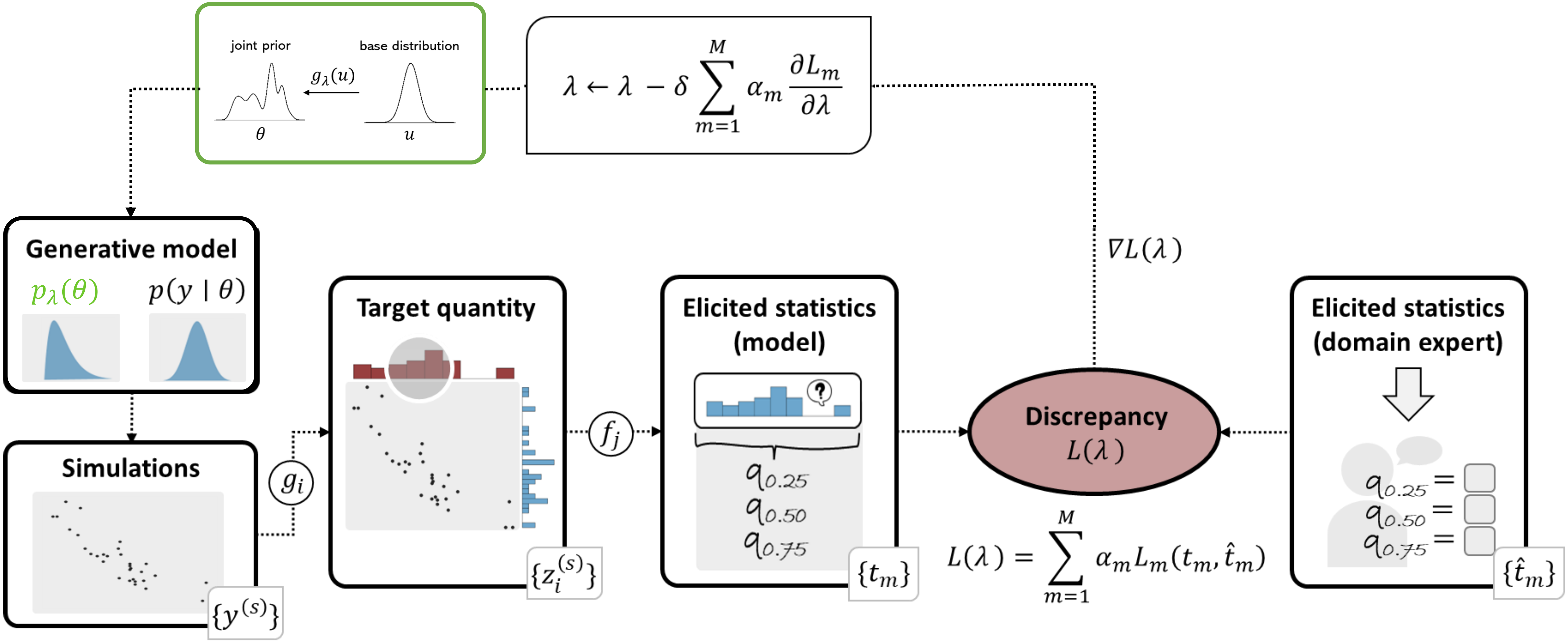
$$\mu_i = \beta_0 + \beta_1 x_{1,i} + \beta_2 x_{2,i}$$

$$y_i \sim \text{Normal}(\mu_i, \sigma\ )$$

florence.bockting@tu-dortmund.de

… what we are doing

Bockting, Radev, & Bürkner (2024; in preparation)



Why is this of interest?

Advantages of learning a joint prior

- ✓ analytic joint prior density for follow-up inference
- ✓ arbitrarily complex joint prior / marginals *(prevent misspecifications in model building)*
- ✓ allows for correlation between model parameters *(increase modelling flexibility)*

Learned via Normalizing Flows

$$(\beta_0, \beta_1, \beta_2, \sigma) \sim p_\lambda(\cdot)$$

$$\mu_i = \beta_0 + \beta_1 x_{1,i} + \beta_2 x_{2,i}$$

$$y_i \sim \text{Normal}(\mu_i, \sigma)$$

Example Revisited: Generate synthetic data

Generative model

(assume independence)

$\boldsymbol{\theta}$

$$(\beta_0, \beta_1, \beta_2, \sigma) \sim p_\lambda(\cdot)$$
$$\mu_i = \beta_0 + \beta_1 x_{1,i} + \beta_2 x_{2,i}$$
$$y_i \sim \text{Normal}(\mu_i, \sigma)$$

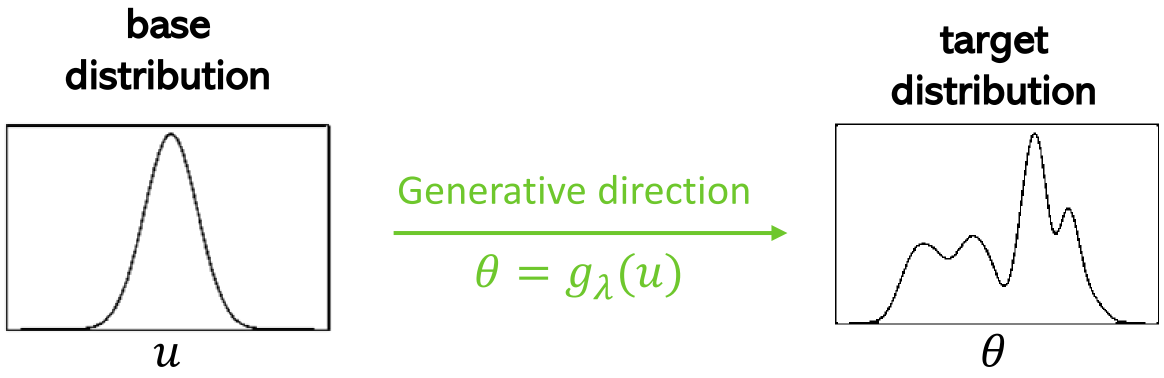
florence.bockting@tu-dortmund.de

A closer look

Sample from joint prior (Normalizing Flow)

Generative model
(assume independence)

$$(\beta_0, \beta_1, \beta_2, \sigma) \sim p_\lambda(\cdot)$$
$$\mu_i = \beta_0 + \beta_1 x_{1,i} + \beta_2 x_{2,i}$$
$$y_i \sim \text{Normal}(\mu_i, \sigma)$$



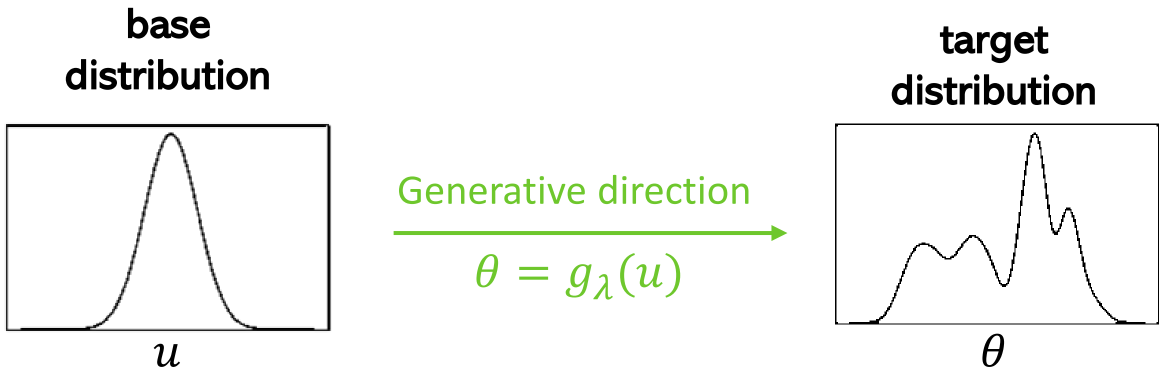
(in reference to Kobyzev et al., 2021)

A closer look

Sample from joint prior (Normalizing Flow)

Generative model
(assume independence)

$$(\beta_0, \beta_1, \beta_2, \sigma) \sim p_\lambda(\cdot)$$
$$\mu_i = \beta_0 + \beta_1 x_{1,i} + \beta_2 x_{2,i}$$
$$y_i \sim \text{Normal}(\mu_i, \sigma)$$



g: *affine coupling flow*

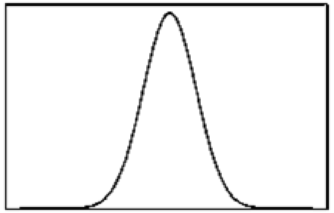
A closer look

Sample from joint prior (Normalizing Flow)

Generative model
(assume independence)

$$(\beta_0, \beta_1, \beta_2, \sigma) \sim p_\lambda(\cdot)$$
$$\mu_i = \beta_0 + \beta_1 x_{1,i} + \beta_2 x_{2,i}$$
$$y_i \sim \text{Normal}(\mu_i, \sigma)$$

base distribution

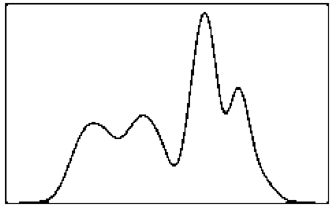


u

Generative direction

$\theta = g_\lambda(u)$

target distribution



θ

g: *affine coupling flow*

Increase expressivity of affine coupling flows

$$g = g_{\lambda_K,K} \odot \cdots \odot g_{\lambda_1,1}$$

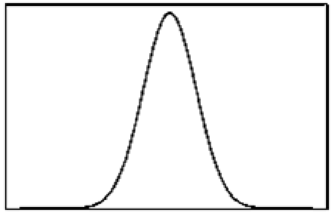
A closer look

Sample from joint prior (Normalizing Flow)

Generative model
(assume independence)

$$(\beta_0, \beta_1, \beta_2, \sigma) \sim p_\lambda(\cdot)$$
$$\mu_i = \beta_0 + \beta_1 x_{1,i} + \beta_2 x_{2,i}$$
$$y_i \sim \text{Normal}(\mu_i, \sigma)$$

base distribution

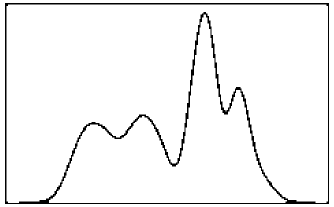


u

Generative direction →

$\theta = g_\lambda(u)$

target distribution



θ

g: *affine coupling flow*

Increase expressivity of affine coupling flows

$$g = g_{\lambda_K,K} \odot \cdots \odot g_{\lambda_1,1}$$

Sample from target distribution

$$\theta = g_{\lambda_K,K}\Big(\ldots\big(g_{\lambda_1,1}(u)\big)\Big), \qquad u \sim p_U(u)$$

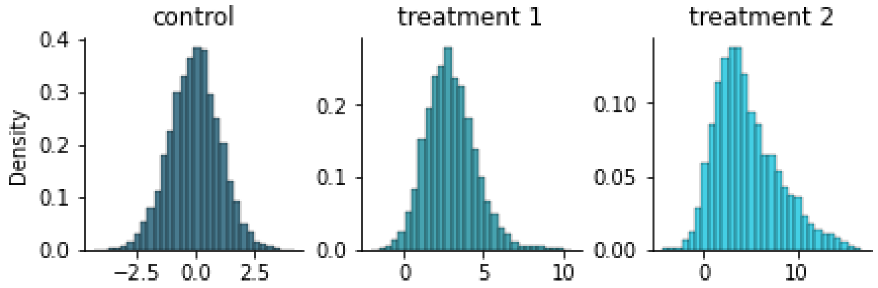
A closer look

Sample from generator & compute elicited statistics

Generative model

(assume independence)

$$(\beta_0, \beta_1, \beta_2, \sigma) \sim p_\lambda(\cdot)$$
$$\mu_i = \beta_0 + \beta_1 x_{1,i} + \beta_2 x_{2,i}$$
$$y_i \sim \text{Normal}(\mu_i, \sigma)$$



$y^{(s)}_{gr,i}$

florence.bockting@tu-dortmund.de

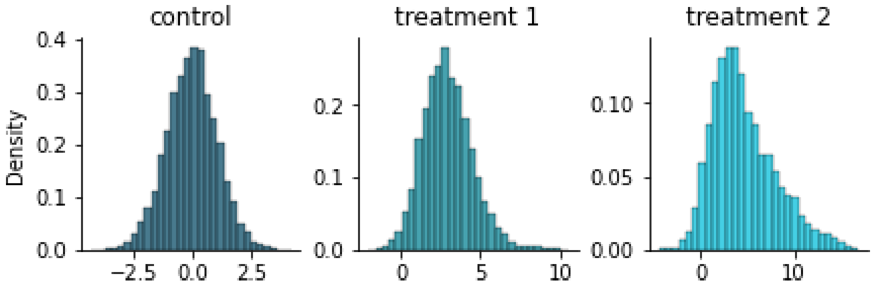
A closer look

Sample from generator & compute elicited statistics

Generative model

(assume independence)

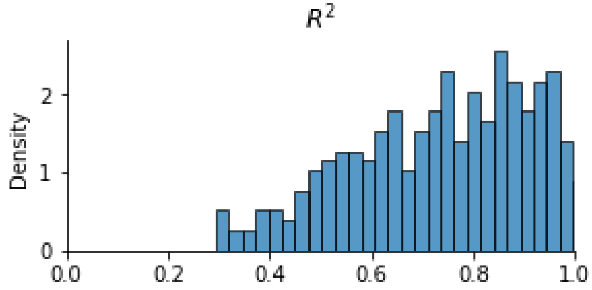
$$(\beta_0, \beta_1, \beta_2, \sigma) \sim p_\lambda(\cdot)$$
$$\mu_i = \beta_0 + \beta_1 x_{1,i} + \beta_2 x_{2,i}$$
$$y_i \sim \text{Normal}(\mu_i, \sigma)$$

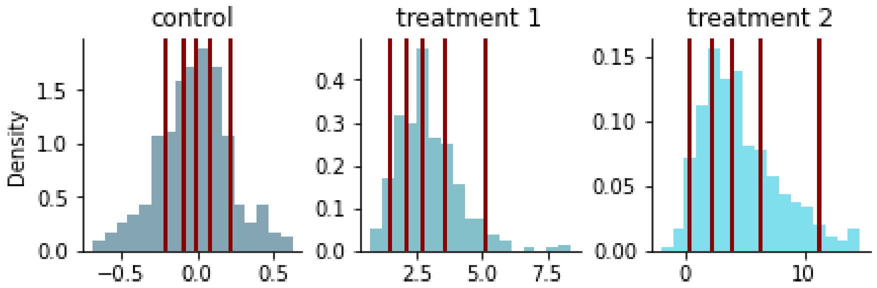


$y_{gr,i}^{(s)}$

Target quantities and elicitation techniques

- Group means: quantile-based
- $R^2 = \frac{\text{Var}(\mu_i)}{\text{Var}(y_i)}$: histogram-based





$\bar{y}_{gr}^{(s)}$

florence.bockting@tu-dortmund.de

A closer look

Compute discrepancy & update weights of NF

- **Compute loss based on simulated data and expert expectations**

$$L(\lambda) = \alpha_1 L_1\left(q_{crt}^{(s)}, q_{crt}\right) + \alpha_2 L_2\left(q_{trt1}^{(s)}, q_{trt1}\right) + \alpha_3 L_3\left(q_{trt2}^{(s)}, q_{trt2}\right) + \alpha_4 L_4(R^{2(s)}, R^2)$$

A closer look

Compute discrepancy & update weights of NF

- Compute loss based on simulated data and expert expectations

$$L(\lambda) = \alpha_1 L_1\left(q_{crt}^{(s)}, q_{crt}\right) + \alpha_2 L_2\left(q_{trt1}^{(s)}, q_{trt1}\right) + \alpha_3 L_3\left(q_{trt2}^{(s)}, q_{trt2}\right) + \alpha_4 L_4(R^{2(s)}, R^2)$$

- Compute gradient of loss w.r.t. λ and adjust λ in the opposite direction of the gradient

$$\lambda \leftarrow \lambda - \delta \sum_{m=1}^{M} \alpha_m \frac{\partial L_m}{\partial \lambda}$$

A closer look

Compute discrepancy & update weights of NF

tu dortmund university
computational statistics

- Compute loss based on simulated data and expert expectations

$$L(\lambda) = \alpha_1 L_1\left(q_{crt}^{(s)}, q_{crt}\right) + \alpha_2 L_2\left(q_{trt1}^{(s)}, q_{trt1}\right) + \alpha_3 L_3\left(q_{trt2}^{(s)}, q_{trt2}\right) + \alpha_4 L_4(R^{2(s)}, R^2)$$

- Compute gradient of loss w.r.t. λ and adjust λ in the opposite direction of the gradient

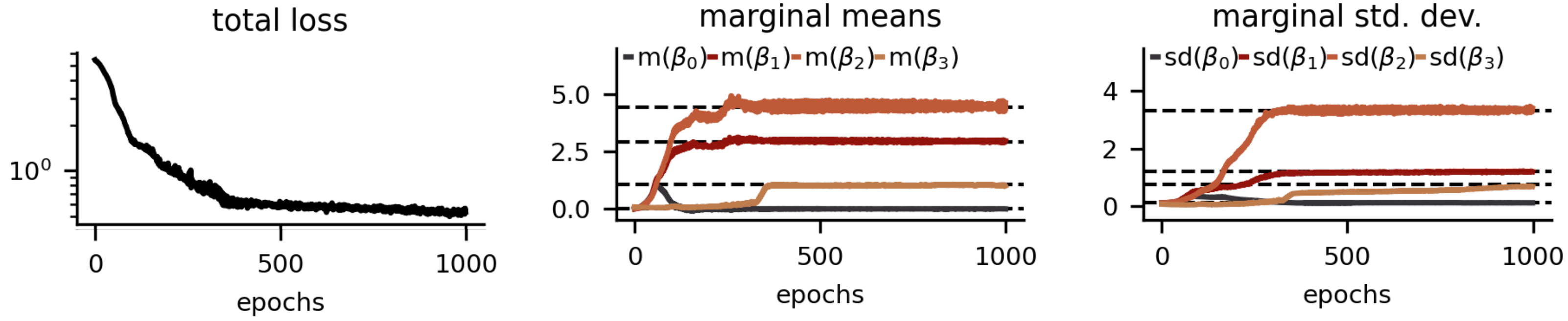
$$\lambda \leftarrow \lambda - \delta \sum_{m=1}^{M} \alpha_m \frac{\partial L_m}{\partial \lambda}$$

- Repeat until max. number of epochs

$$\text{update}(\lambda^{t_0}) \mapsto \lambda^{t_1}$$

...

$$\text{update}\left(\lambda^{t_{\max-1}}\right) \mapsto \lambda^{t_{\max}}$$



florence.bockting@tu-dortmund.de

A closer look

Faithfulness: Learned vs. true elicited statistics

Generative model

(assume independence)

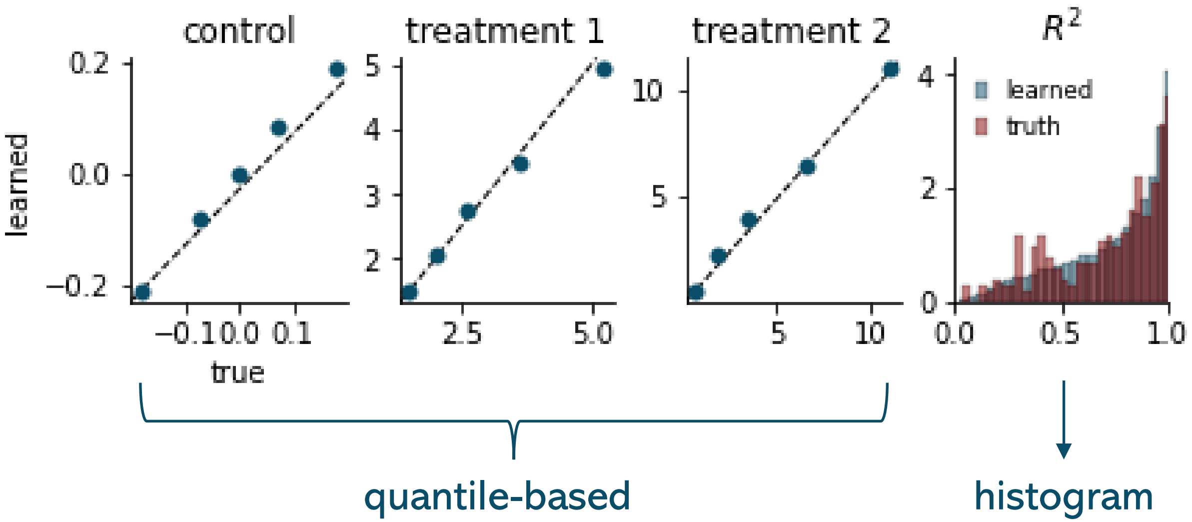
$$(\beta_0, \beta_1, \beta_2, \sigma) \sim p_\lambda(\cdot)$$
$$\mu_i = \beta_0 + \beta_1 x_{1,i} + \beta_2 x_{2,i}$$
$$y_i \sim \text{Normal}(\mu_i, \sigma)$$

Ground truth

$$\beta_0 \sim \text{Normal}(0., 0.1)$$
$$\beta_1 \sim \text{SkewNormal}(1.5, 0.3, 6.)$$
$$\beta_2 \sim \text{SkewNormal}(1.5, 0.8, 6.)$$
$$\sigma \sim \text{Gamma}(2., 2.)$$

Target quantities and elicitation techniques

- Group means: quantile-based
- $R^2 = \frac{\text{Var}(\mu_i)}{\text{Var}(y_i)}$: histogram-based



A closer look

Learned joint prior

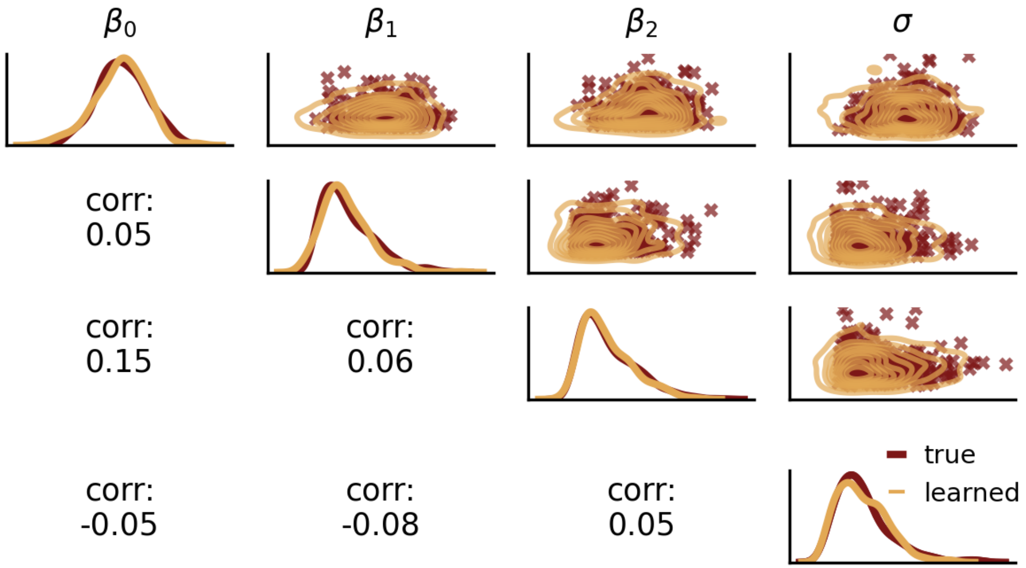
Generative model

(assume independence)

$$(\beta_0, \beta_1, \beta_2, \sigma) \sim p_\lambda(\cdot)$$
$$\mu_i = \beta_0 + \beta_1 x_{1,i} + \beta_2 x_{2,i}$$
$$y_i \sim \text{Normal}(\mu_i, \sigma)$$

Ground truth

$$\beta_0 \sim \text{Normal}(0., 0.1)$$
$$\beta_1 \sim \text{SkewNormal}(1.5, 0.3, 6.)$$
$$\beta_2 \sim \text{SkewNormal}(1.5, 0.8, 6.)$$
$$\sigma \sim \text{Gamma}(2., 2.)$$



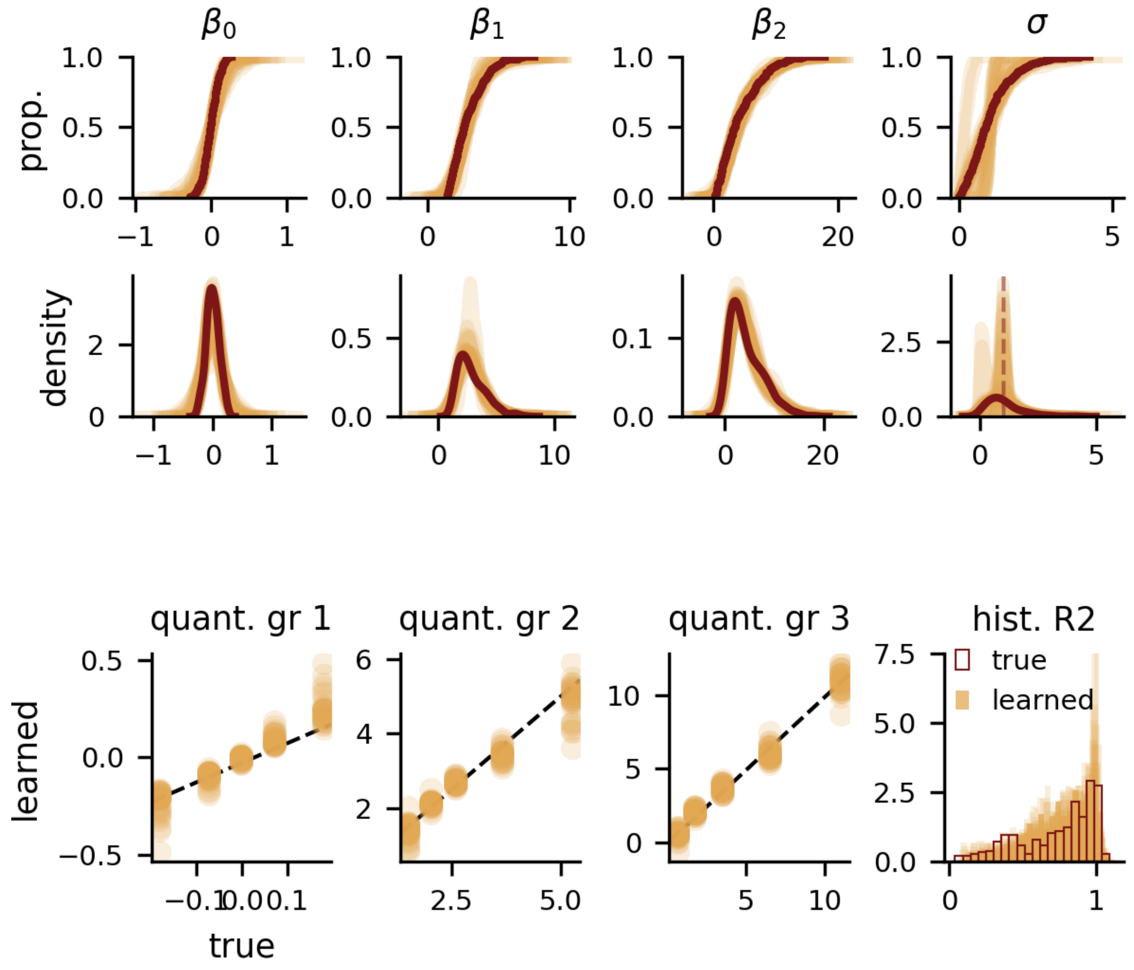
florence.bockting@tu-dortmund.de

Sensitivity Analysis

Uniqueness and Faithfulness

dortmund university
computational statistics

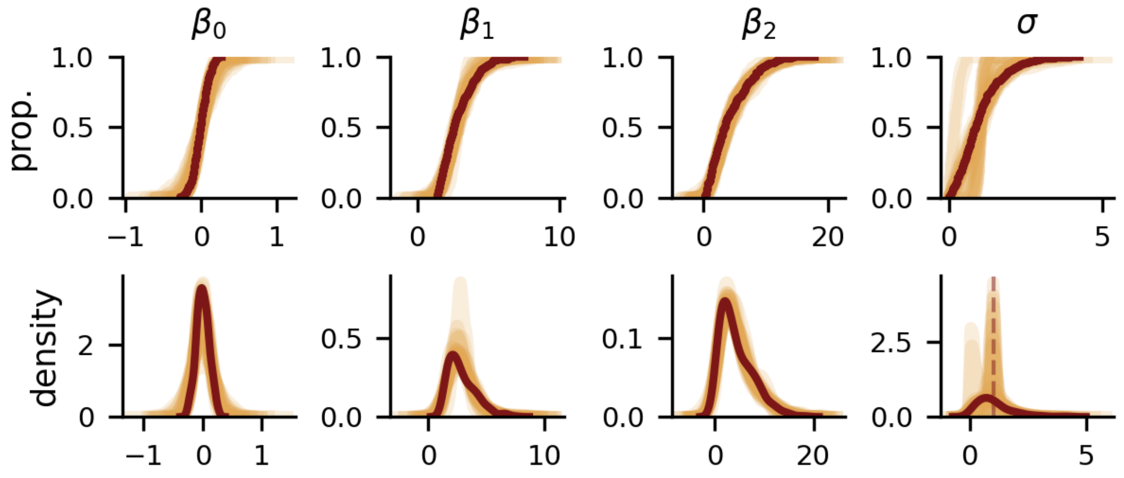
- 30 independent replications with different random seed but same true data
- Elicited statistics are learned accurately
- Joint prior for provided elicited statistics is not unique



Sensitivity Analysis
Uniqueness and Faithfulness

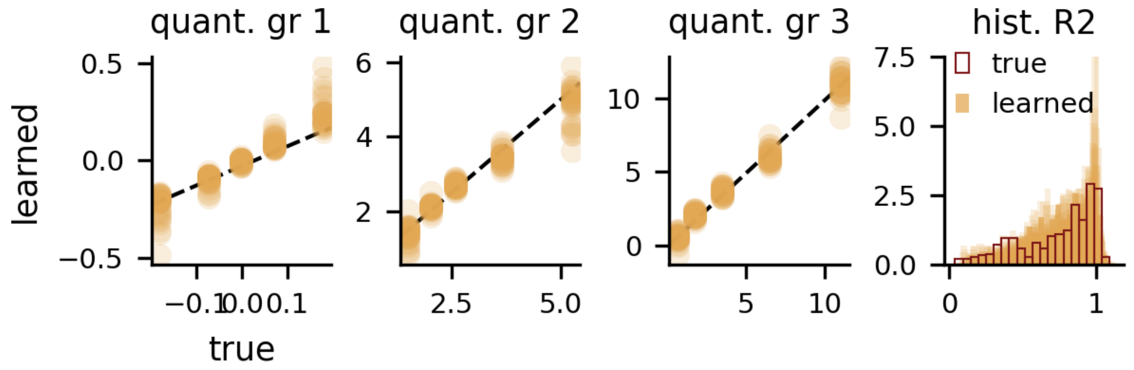
tu dortmund university
computational statistics

- 30 independent replications with different random seed but same true data
- Elicited statistics are learned accurately
- Joint prior for provided elicited statistics is not unique



Dealing with non-uniqueness:

- Elicit additional information from the expert
- Select one plausible joint prior
- Prior averaging



florence.bockting@tu-dortmund.de

Outlook

Where to go from here?

- Approaches that deal with multiple expert beliefs
- Interface to R/Stan (current implementation is in Python TensorFlow)
- Tutorial paper for practitioners (incl. ‘good’ diagnostics, default values for minimizing tuning, standard workflow, etc.)
- Applications

florence.bockting@tu-dortmund.de

dortmund university

computational statistics

Thank you for your attention.

Contact:

Florence Bockting
TU Dortmund University, GER

florence.bockting@tu-dortmund.de

Stefan T. Radev
Rensselaer Polytechnic Institute, NY, USA

radevs@rpi.edu

Paul-Christian Bürkner
TU Dortmund University, GER

https://paul-buerkner.github.io/

tu dortmund university
computational statistics

Explicit form of the target distribution by change of variables formula

$$p(z) = p_Z(z)$$

$$p(\theta) = p\big(z = g(\theta)\big)\det\left(\frac{\partial}{\partial\theta}g(\theta)\right)$$

Obtain samples from $p(\theta)$

$$\theta = g^{-1}(z) \sim p(\theta) \text{ for } z \sim p_Z(z)$$

Model g as affine coupling flow

$$v_1 = u_1 \odot \exp\big(s_1(u_2)\big) + t_1(u_2)$$
$$v_2 = u_2 \odot \exp\big(s_2(v_1)\big) + t_2(v_1)$$

With inverse g^{-1}:

$$u_2 = \big(v_2 - t_2(v_1)\big) \odot \exp\big(-s_2(v_1)\big)$$
$$u_1 = \big(v_1 - t_1(u_2)\big) \odot \exp\big(-s_1(u_2)\big)$$

Input vector: $u = (u_1, u_2)$ with $u_1 = u^{(1:d)}$ and $u_2 = u^{(d+1:D)}$